DOCUMENT RESUME

ED 234 756 IR 010 822

AUTHOR Siegel, Martin A.; Davis, Dennis M.
TITLE The PCP SYS IV Management System: Educational Overview.
INSTITUTION Illinois Univ., Urbana. Computer-Based Education Research Lab.
REPORT NO CERL-E-27
PUB DATE May 83
NOTE 25p.; For related documents, see IR 010 822-824.
PUB TYPE Reports - Descriptive (141)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; *Computer Managed Instruction; *Computer Programs; *Design Requirements; Futures (of Society); *Individualized Instruction; Programing Languages; Recordkeeping; Specifications
IDENTIFIERS Computer Uses in Education; Instructor Logic Programing Language; Interactive Systems; *PLATO

ABSTRACT

This paper describes the structure and function of the SYS IV PLATO Curriculum Project's (PCP) computer-based management system and justifies the system design. A consideration of features most critical in the design of management systems provides the context for a discussion of specific SYS IV features, which include its ability to deliver any type of lesson in a variety of different management configurations; capacity for contingent branching; division into a fixed and variable part, which makes possible its flexibility and ease of use; and its unique methods for presenting, summarizing, and storing data in varied formats. Design considerations are addressed as a set of tradeoffs: power versus wide applicability, flexibility versus ease of use, and amount versus accessibility of data. The technical means by which features are provided are explained, including the programing language, Instructor Logic (IL), and the SYS IV options of index, topic size, instructional type, assignment, saving data, learning path, instructional intervention, delays, records options, and the implementation scan utility, which groups data for reports, tabulations, and summaries. A conclusion summarizes system advantages and the need for "computer imagination" to actualize the potential for computers in education. (LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

- This document has been reproduced as received from the person or organization originating it.
- Minor changes have been made to improve reproduction quality.
- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

Computer-based Education Research Laboratory

CERL

ED234756

THE PCP SYS IV MANAGEMENT SYSTEM: EDUCATIONAL OVERVIEW

MARTIN A. SIEGEL

DENNIS M. DAVIS

CERL REPORT E-27

MAY 1983

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
D. Bitzer

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

University of Illinois at Urbana-Champaign

CE010822

Copyright © May 1983
by the Board of Trustees, University of Illinois

All rights reserved. No part of this book may be reproduced in any form or by any means without permission in writing from the authors.

The PLATO® system is a development of the University of Illinois.

PLATO® is a service mark of Control Data Corporation.

Acknowledgement

The PCP SYS IV Computer-based Management System was developed by Martin A. Siegel, Project Director; J. Michael Felty; John T. Bryan; Stephen M. Alessi; Lindsay P. Reichmann; Walter G. Brooks; David A. Fuller; John B. Gilpin; and Peter K. Zelchenko.

This paper is the first in a series on SYS IV. In the other two papers, John T. Bryan and J. Michael Felty present a technical overview, and Dorothy Z. Silver and Elizabeth J. Clapp outline an implementation and training overview.

THE PCP SYS IV MANAGEMENT SYSTEM: EDUCATIONAL OVERVIEW

Martin A. Siegel
Dennis M. Davis

PLATO Education Group
Computer-based Education Research Laboratory
University of Illinois at Urbana-Champaign

The bewildering number of approaches to computer-managed instruction and to the structure and function of computer-based management systems make one fact plain: there is no such thing as "the perfect management system." That statement is true in a literal sense. It's not just that nobody has built the perfect system; it's that there can be no single system that's perfect for all uses. Even the "perfect" management system for a discovery-based computer learning environment, for example, would be woefully inept at managing so-called "traditional" computer-assisted instruction. The excellence of a management system, in other words, depends in the first place upon (should, in fact, be determined by) the instructional approach of the courseware it delivers.

This paper, therefore, has two purposes. On the one hand, it describes the structure and function of the PLATO Curriculum Project's (PCP) computer-based management system, SYS IV. Given the sentiments expressed in the preceding paragraph, however, its additional function is to justify the design of that system. Our task is not merely to explain what SYS IV does, but also to explain why. For that reason, we shall discuss briefly our approach to instruction and the interrelationship we feel should exist between instruction and management. Within this context, the considerations we feel are most critical in the design of management systems will become apparent. And within that context, those features of SYS IV, which we will discuss subsequently and which we think are most unique and worthy of interest, take their meaning.

Thus, we begin with the following instructional premises:

1. The best use of computers in education is as a transparent medium that delivers instruction.

   The computer has been put to a wide variety of uses in education. Taylor (1980) classified them into three broad categories: tutee (learning through programming, that is, by teaching the computer to do things), tool (applying software features like super-calculation to instructional tasks), and tutor (using the computer to deliver instruction in academic subjects). Of these, we believe that the tutor mode is "best"--not because it is intrinsically superior, but because it is the mode best capable of addressing the most pressing needs of American education most directly and effectively.

Data, like declining scores on standardized achievement tests, and studies, like the recent report of the National Commission on Excellence in Education show that what we critically need is increased instructional effectiveness and efficiency. Few doubt the computer's potential contribution to the remediation of this problem. While all the approaches to CBE Taylor mentioned have value and should be pursued, we believe that tutor-style CBE is the best of available options for tackling problems of instructional effectiveness on a large scale with the expediency demanded by the nature of our educational system's current needs.

2. Effective CBE of the tutor style must be fully individualized for maximal effectiveness.

We believe that individualizing instruction is a key factor in increasing effectiveness. We also believe that a chief strength of the computer as an educational medium is its capacity to individualize instruction. Instructional individualization is a concept that can be defined and achieved in many different ways. In practical terms, we see it as a simple tradeoff:

- Individualization depends on maximizing fruitful instructional interaction.

- Because they can't think, computers are not as good at interaction as human teachers on a one-to-one basis, but, because of their other strengths (recordkeeping, branching, the fluid and dynamic ways in which they can store and present information, etc.) they have the potential to be better at interaction when instruction must be delivered to a classroom of several students.

The key to excellent tutor-style courseware, then, lies in effective exploitation of the computer's interactive capacities, and the contribution of a management system to this instructional goal is extremely important, as we shall see.

3. The PCP definition of individualized instruction is linked to mastery.

In fact, linking individualization to mastery enables a clean, useful, operational definition of individualized instruction: instruction is individualized when each student receives all instruction he needs to achieve mastery on the skill or concept being taught, and no student receives more instruction than he needs to achieve mastery.

4. Individualization resides in management as much as in instructional lessons.

If individualization depends on interaction, within a lesson the courseware will behave differently, depending upon student input in the form of answers to questions and performance on skills. The nature of that sort of interaction is direct--the student does something and the courseware responds contingently. However, CBE that individualizes instruction fully must also perform interactively on tasks external to the lessons themselves--tasks like placing students at the correct starting point for instruction, branching them to subsequent instruction after the lesson on the basis of their performance either in the lesson or on a posttest, providing distributed practice for long-term retention and transfer of training, etc. This set of individualizing aspects of an instructional system depends on indirect interaction--the system does something contingent upon data it collects on student performance on tests and in instruction, not on direct student input.

Adapting instruction to the individual on the basis of indirect interactions like these is the proper area of activity for a computer-based managment system. When a management system is conceived as an integral part of a fully individualized instructional system that blends testing and instruction, the effect is synergistic: the effects of the separate components--testing, instructional curricula, and management--are each enhanced.

5. To best achieve individualization, CBE should be autonomous.

Unqualified, a statement like this can seem disquieting for classroom teachers, if not downright threatening. Yes, autonomy does mean that the system--instructional lessons delivered by a management system--should require a minimum of human teacher intervention. But no, the intent of striving for autonomy is not to replace the classroom teacher.

Rather than obviating the contribution of the classroom teacher, an autonomous instructional system provides the teacher with a guarantee of mastery. In other words, when students return from work "on the computer" in some subject, the teacher knows just what has been taught while she "wasn't looking"--how the student is thinking about the concepts he learned and the skills he acquired, and, most important, that the student has--in fact and for certain--actually mastered the instruction for which he was remanded to the computer.

6. Autonomy depends on the availability of a sophisticated management system.

The management system that delivers instruction autonomously must be structured to keep data, not just on a student's performance, but also on his activities while working on the computer, his path through instruction, his problem areas, etc.--and all of it in as much detail as the individual instructional task and individual student warrant. To accomplish this objective, a management system must be powerful, but it must also be flexible. It must be adaptable to a wide variety of lesson types and instructional situations. In addition, if it is to be effective in the hands of a teacher who is not a computer expert, it must somehow offer that kind of power and flexibility in a system that is easy to use. In other words, a management system of the highest possible level of sophistication is an indispensable part of a computer-based instructional system that takes as its goal the delivery of fully individualized instruction that guarantees mastery.

Design Considerations of Managment Systems

In designing any management system--or in choosing one for implementation--one's point of view on the goals of instruction, the pedagogical methods for achieving them, and the idea of individualization are the primary considerations, because, as we have suggested in the preceding section, they determine the definition of "management" embodied in any system. Even within a single perspective on instruction and management, however, a large number of different configurations for computer-based systems exists. There are, in other words, additional design considerations that operate at the level of the individual system, and, given the instructional point of view within which the system operates, these design features determine the level of sophistication of the system--and, indeed, whether it is good or bad. Again, they are best expressed as a set of tradeoffs:

Power vs. Wide Applicability

Some management systems are designed to run courseware programmed expressly for them. The advantage to this configuration is that instructional lessons save and transfer a great deal of precise data on student performance to the management system, maximizing its power and subtlety. The flexibility of this type of system, however, is severely limited--its range of applicability is restricted--because only lessons programmed to run on it can be managed.

The way to increase flexibility, of course, is to detach management functions from the courseware--make the system independent of the lessons it delivers. Unfortunately, some management systems that attempt this lose too much power. These systems amount to computer-managed lesson indices.

The aim of SYS IV is to make available for delivery essentially any lesson in the PLATO catalog. Some of the lessons it uses are PCP lessons, especially programmed to save data and transfer them to the management system, but any TUTOR lesson can run in SYS IV effectively, because the system is uniquely designed in two parts. The "constant" part, which manages roster and usage data for individual students and groups of students, operates independently of instruction. The "variable" part of the system manages instruction---assigning it, testing, and branching contingent upon student performance, etc.

Flexibility vs. Ease of Use

Management systems of the first kind described above are usually easy to use, since the lessons and the management work closely together. Often, when a more "general purpose" or flexible system is developed, one of two things happens. Sometimes the number or variety of management options is restricted to certain standard, commonly requested features, which reduces flexibility but assures ease of use. Sometimes a large number of features are included for flexibility, but then the system often becomes difficult or tedious (or both!) for the user: the software equivalent of a vast bank of dials and switches that must be set for each lesson. The trick, in designing the system, is to somehow keep the widest possible variety of user options available while ensuring that the system will be easy to use.

Amount vs. Accessibility of Data

Integral to the utility of a sophisticated management system is the automatic storage of extensive data on users: administrative data and data on instructional performance. Too much stored data, however, cause problems. It is not long before the usefulness of data is lost in the deluge.

The key to establishing balance is realizing that it's not the amount of data being kept that causes problems, but the length of time it is kept and the form in which it is kept. Detailed information on student activity and performance is of great short-term use to the instructor, but it becomes much less useful as it ages. Management data, then, have a sort of "half life" of utility--their utility decays as a function of time. Often, data continue to have usefulness, but not in raw form; they require summary, averaging, and tabulation to retain utility. Thus, a solution that simply lets data "fall off the edge" is a poor solution to the problem of amount vs. accessibility.

In addition, different kinds of users require access to different amounts of data, and they need to see it in different forms. Students, in general, need not deal with much of the data kept on their performance. Sometimes making it available is useful (for example, as a motivator or reinforcer), but often it interferes. Instructors, as we have said, need detailed data on students, presented in a form that facilitates close monitoring of student activity. Administrators (and instructors, when they perform administrative tasks) need either

summarized data, or data tabulated in different ways. Finally, the needs of summative instructional evaluators and the developers of the instruction, who want to make formative evaluations, are different still. A sophisticated management system like SYS IV provides for all of these users--not just keeping but also managing data for greatest utility to each type of user.

In summary, in addition to reviewing the principal design considerations of management systems that aim at autonomous delivery of fully individualized instruction that guarantees mastery, we have indicated three design features that set SYS IV apart from other management systems:

1. its ability to deliver any type of lesson (or any combination of lessons and tests) in a variety of different management configurations, combined with a capacity for contingent branching;

2. its unique division into a fixed and a variable part, which makes possible the flexibility and ease of use mentioned above; and

3. its unique methods for presenting, summarizing, and storing data in formats of use to a variety of users.

Next, we shall discuss in detail the technical means by which these features are provided.

Instructor Logic

The variable part of SYS IV--the part that manages instruction and permits the system to deliver any type of lesson or instructional configuration--is variable because of a feature called "Instructor Logic" (IL). IL is a high level programming language, in some respects similar to TUTOR (naturally, since it runs on PLATO), whose commands are instruction-related. Among them, for example, are -assign-, -unit-, -test-, -index-, and -branch-. The language is sophisticated and powerful, but the context in which it functions is equally elegant.

In effect, the part of SYS IV that manages instruction (as opposed to the fixed part that manages roster and usage operations), is fully programmable. The system, instead of "wiring in" a set--large or small--of "features" at the beginning, has been designed so that in the part where variability is important (variability that occurs from different instructional needs for different curricula managed, and different lesson configurations), the system is completely soft. IL, thus, is a language that bends TUTOR to the kinds of purposes building instructional management requires.

This design characteristic of SYS IV, of course, accounts for its flexibility. But there's more. The way IL is implemented in the system is also important. We mentioned that a drawback of the "feature"

approach is that users have to face a bewildering array of options, choosing among them, turning them on and off for each lesson. The advance that IL makes in user-friendliness is that it operates at a level just behind the user's. The instructor is not required to master IL to use SYS IV; the "programming" of instructional management for the curricula he wishes to deliver is done for him by programmers. The user interface requires no programming expertise. Adding, deleting, or changing the delivery mode of curricula for a school or other user group is a simple matter of rewriting the IL for their copy of SYS IV to suit their new or additional demands. It's done once, in a matter of a few hours. In short, one aspect that makes SYS IV sophisticated is the way it clearly distinguishes user and programmer.

In addition, the same design characteristic that makes SYS IV easily adaptable to the needs of individual users also enables the system to grow or change without effort. If a new user need arises for which the current IL is insufficient, you don't write another management system, you write a new command! SYS IV is, as its name suggests, the fourth "edition" of PCP management systems, the first to incorporate IL. With IL, it will probably be the last. From this point on, changes can be made internally without having to redesign the entire system.

Finally, IL is innovative in another, more subtle way. The current rage for micros operates on the assumption (seldom realized by educational users and almost never expressed by anyone, including CBE experts) that computer-based education is deliverable "whole"--in the same way that texbook curricula are. Microcomputers are set up to deliver education on disks that schools buy like books and then just use. Anyone familiar in the least with CBE (or with software in general) realizes that such a naive expectation is unrealistic. Large software systems (including sophisticated management systems) can't "just be used." They must be implemented--not simply because all software has bugs, but also because it must be maintained, and because the dynamic nature of computer use requires constant change. Thus, SYS IV is a management system that has been designed and programmed not to be a "definitive statement" but to have the capacity for change incorporated as an integral part of the design of the system.

SYS IV Options

Within the programmable part of SYS IV--the section that manages instruction and is governed by Instructor Logic--the following system options are available. These options are not under direct instructor control but are programmed for each curriculum when it is added to the system:

1. Index. The system provides the option either of making an index (menu) of lessons available to the student or of setting up the instructional sequence, so that the student must go through it in a prescribed order.

2. Topic size. A single topic in the student's catalog may range from a single lesson or test to a whole integrated curriculum.

3. Instructional types. The system is capable of delivering any combination of pretests, instructional lessons, and posttests.

4. Assignment. The system permits the choice of allowing students to reenter lessons as many times as they like, or of causing the titles of completed topics (and thus the student's access to them) to disappear from the student's index. Such topics can also be made to reappear according to programmable criteria. (For example, if a game lesson is used as a reinforcer, it can be made available according to criteria of time or performance.)

5. Saving data. Lessons can be especially programmed or adapted to transfer data on performance to SYS IV, or existing lessons can be used as they are, and SYS IV will get whatever data are available (for example, restart or scoring information saved in -lstatus-).

6. Learning path. A student's path through a curriculum set up as a SYS IV topic can be straight, or it can be programmed by IL to be contingent on performance or other criteria.

7. Instructor intervention. IL also provides the capability within a topic for flagging the instructor for any reason, including software problems, a test score too close for the machine to call, or at a point at which offline instruction is programmed into the curriculum. (Thus, while SYS IV is designed to operate with full autonomy, it is easily adaptable to a mixture of on- and offline instruction with instructor intervention.)

8. Delays. IL's capacity to program in delays (in minutes or in days) is an important feature in a fully autonomous system. Among other things, it provides for distributed practice without the need for instructor intervention.

The previous set of variable instructional options, controlled by the IL, is invisible to the user. The options are programmed into the management system in a configuration chosen and especially adapted to the curricula the system is delivering. The fixed group of options with which the user (instructor, student, administrator, evaluator, or researcher) interacts directly, of course, is visible. Because the instructor manipulates them directly, the most useful format in which to discuss them is that in which they are presented to the user--beginning with the instructor, who gets the most detailed information from the system.

Figure 1 represents the screen display presented to instructors when they first enter SYS IV, and it reflects instructor options at the most general level. Note in the figure that instructors have full access to notes and records (that is, they may input data as well as read them),

but they have inspect privileges only for the SYS IV Topic Catalog and for the main PLATO catalog. The reason for restricted access to the instructional lessons is that these are governed by the IL. From their level, instructors may see the instruction that IL is delivering, but if they want things changed, they need to go to the next deeper layer of the system, as we described above.

Insert Figure 1 about here

Notes. Communications are an important part of a fully autonomous management system, especially when users "implement" an instructional system rather than merely "consume" it. Several different communications are necessary in an implementation situation: instructor to student, instructor to instructor, instructor to system staff, etc. For this reason, SYS IV makes several different kinds of communications available to users. First, it utilizes the PLATO notes systems for instructors, allowing access to personal notes, some notesfiles, and to a sequencer at this option. In addition, of course, SYS IV has TERM-comment available, with comments stored in PCP notesfiles for maintenance and instructional problems. Finally, instructors may leave signon messages for individual students or for a whole section. This feature is available under the records options discussed below.

Catalog. In order to accommodate a wide variety of instructional types, SYS IV employs a hierarchical structure for organizing lessons. The catalog is divided into subject areas, like math, language skills, and reading comprehension. Subject areas are divided into topics. The topic is the level that shows in a student's index as a single entry. Structurally, however, topics may be comprised of single lessons, like addition of fractions, multiplying mixed numbers or they may be curricular sequences, like basic arithmetic or GED reading. In other words, topics may range in size from a single unit to an entire curriculum comprised of many instructional units sequenced together under a single index entry.

Units are built in IL, and they may include both online and offline work, both PCP courseware or any other PLATO courseware, and they may deliver any combination of tests, assignments, and instructional configurations. As an example, one unit structure frequently employed is shown in Figure 2, but virtually any other configuration is possible. Nevertheless, this simple structure, comprised of pretest, and/or an instructional lesson, and/or a posttest, suffices to show some of the flexibility of SYS IV which results from the IL:

1. If a pretest is given and the student fails it, the instructor may intervene to allow the student to retake the test, make a special assignment, or, if the student's performance is in doubt, permit him to go to the longer, more comprehensive posttest of the unit to get more information about the student's mastery level. (If the student passes the pretest in a multi-unit topic, he goes on to the next unit's pretest, skipping the instruction in that unit.)

2. After the instructional lesson is completed, the system permits a pause for completion of offline work, or it allows the student to go right on to the posttest.

3. If the student passes the posttest, the system routes him to the next unit in the topic. If he fails the posttest, it permits instructor intervention to readminister the posttest, assign additional, offline instruction, or route the student back through the lesson again. (Lessons can be structured to include a review mode that, among other options, may permit the student to start in the middle of the lesson after the initial teaching sequence.)

Insert Figure 2 about here

The catalog instructor option permits instructors to inspect topics at a variety of levels of specificity. First, the system presents the list of subject areas. Choosing one, the instructor is presented with a list of topics in that subject area. Choosing a topic, he is given first a prose description of the topic--the skills taught, the number of units in the topic, the number of pre- and posttests and their placement within the topic, etc. If he wants more information, the instructor may then view the units as the student sees them.

Records. This option represents the main center of activities for SYS IV instructors. The records that instructors manipulate (inspect or change) at this option fall into two main categories: administrative functions and instructional functions. Figure 3 shows the screen display that plots when the instructor chooses option "r" from the initial display shown in Figure 1. The functions available from this display are administrative in nature, and their natures are self-evident from the figure.

Insert Figure 3 about here

When the instructor presses "r" and a student's roster number from the display shown in Figure 3, an individual student's current record appears (see Figure 4). The list at the top of the record represents the index of topics from the SYS IV catalog currently assigned to the student. The options available below the box in the center of the display provide information--administrative and instructional--on the student. The following instructional options are available:

1. General information includes the student's starting date in SYS IV, the time he was last on, his total hours on PLATO, total time in instruction, and his time per day in the system.

2. The topic summary includes the date each topic was assigned, days the student has spent in the topic, his current unit, days in that unit, and total instructional time spent in that unit.

3. The single topic option provides more specific data about an individual topic, including such things as pre- and posttest scores. From this option, it is also possible to access even more detailed information about the student's performance in the topic, including such information as the exact test items he got right and wrong, and his individual prescription for this topic.

4. The history option provides similar summary information from topics the student completed in the past and which are no longer in his index.

In addition, the following administrative options are available:

1. The assign option allows the instructor to assign the student a new topic.

2. The copy option allows the instructor to assign a standard prescribed group of topics. This is most useful for a new student.

3. Similarly, the delete and order options permit topics to be deleted and reordered automatically.

4. The note option was discussed above. It allows the instructor to write a signon note to the student.

5. The minutes delay option permits the instructor to specify a period of time after the student signs on each day during which the delayed lessons do not appear in the student's index (and therefore may not be accessed by the student). Thus, in effect, it controls the order in which students move through topics. For example, if the instructor wants a student to work on reading first, he sets a delay (perhaps 15 minutes) on the math topic in the student's index. Another example: a student's time on PLATO is 1 hour a day, and a game is included in his index as a reinforcer. The instructor can set a 45-minute delay on the game, so that the student may spend no more than the last 15 minutes of the period in the game.

Insert Figure 4 about here

The final group of options, in the lower right corner of Figure 4, are the controls that allow the instructor to move back and forth within the current records of students. SYS IV organizes students into sections (which correspond to a classroom organization), sections into groups (which correspond to schools or implementation sites), and groups into a system. In addition to the practical utility of this arrangement to instructors and others who need access to data, it facilitates two other important functions. First, user types (which include system staff,

group directors, instructors, and two types of instructor aides--those with communications privileges and those without) require access to data at different levels of inclusiveness, and the organization of records in this fashion facilitates the management of access. Secondly, data summaries of different kinds for different uses require similarly variable inclusiveness, and the arrangement of records hierarchically facilitates their management as well.

Implementation Scan Utility

Grouping data for reports, tabulations, and summaries is facilitated by the structure of SYS IV user groups, but it is accomplished by the Implementation Scan Utility (ISU). Instructors most often need current and past information on individual students and in raw form, so this is the information immediately accessible from the administrative-instructional managerial (fixed) options discussed above. Sometimes, however, instructors (and other SYS IV users) need summary information, either on an individual student or on groups of students: tabulated, compressed, and/or averaged. These data are not needed as frequently, nor is there a large number of standard configurations in which they are useful. For both these reasons, summary data are accessed through a separate SYS IV utility, the ISU.

The ISU consists of a user interface "lesson" (program) that leads the user through a series of binary choices about student data and, on the basis of these choices, constructs a search algorithm which the user then submits to a runner system that munches stored SYS IV data, organizes it, and returns output to the user either in an online list or print.

In using the ISU, the instructor answers these questions:

1. Which sections should be included?
2. Should the records of students no longer in the section(s) be included in the scan?
3. Should the scan include instructors as well as students?
4. At what date should the scan begin?
5. At what date should it stop?
6. Within this range, what is the range of dates of last use of records that should be included in the scan? (This option allows the user to omit or include inactive signons.)
7. What kinds of information should be included in the output:
   a. basic information (name/group/section and ID number)?
   b. basic information plus general information (see discussion of Records option above)?

c. all of the above plus topic summaries?

d. all of the above plus histories of current topics?

e. all of the above plus histories of deleted topics?

8. Should the print be organized by sections?

9. Do you want to specify particular topics to scan?

   a. The default situation is to scan the specified records for all topics.

   b. A single, individual topic may be specified.

   c. A subset of topics may also be specified, to include "all records that have topic x and topic y" or "all records that have topic x or topic y."

   d. Within the records specified by topic, the user can request data on the specified topics alone, or all data in student records that include the specified topics.

With this utility it is possible to query the SYS IV data stored on student and instructor records and actually get answers to questions like this: "Tell me the number of all students, past or present, in the SYS IV system who, after skipping Units One and Two of the language curriculum, passed the Unit Three posttest in two tries or less. List their names and print their records." The utility is of obvious use to users of all levels and for a variety of purposes. Instructors can request information to be used in making instructional decisions; administrators can easily generate tabulated, summarized, and averaged data (with standard deviations automatically computed, by the way) for reports and checks; researchers can easily request data for studies and formative evaluation; and the task of summative evaluation is largely automated.

Considering its power, the query format presented above is remarkably straightforward. Nevertheless, if a particular data configuration were needed with even infrequent regularity, the task of building the binary retrieval structure each time would be tedious. SYS IV provides for this situation with two additional options. First, a binary of common utility can be named and stored in a file, then requested by anyone who needs it. Thus, for example, the structure for doing a monthly report can be always readily at hand for whoever writes it. Second, if an individual user has a routine of personal utility, it can be stored in a file linked to his signon but accessed through the ISU utility. Thus, it is as readily available as the common store of routines, but only to the individual user.

Conclusion: Good Ideas

The computer in education is not simply a new technology; it is a new educational medium. Those who view it merely as technological advance and fear it may supersede teachers or books--like a new process for making steel replaced an old one, or the overhead projector replaced the opaque projector--miss the point. As a new medium, the nature of its contribution to education is unique, so, rather than supersede, it will take its place beside other proven educational media like teachers and books. That doesn't make it less important or revolutionary--but more so. Many claim, for example, (usually without specifying how) that, in its potential, the introduction of the computer in education is perhaps more important than the invention of the textbook.

That statement may seem extravagant to educators who have idle computers in their schools, or computers used exclusively for entertainment or merely to teach programming. However, it is a characteristic of new media that they must create their own uses, and they do so by reorganizing the cultures into which they are introduced. A new technology suffers from no such growing pains, because it serves an existing purpose: the changes it makes merely enable that which is already done somehow to be done better--more quickly, more easily, more cost-effectively. A new medium, however, becomes "useful" by creating a new dimension in its culture.

Thus, despite the bright promise those who champion educational computing claim for the medium, it is not surprising that the mere acquisition of a few micros by a school is not turning education around. We must pass through a period in which the new medium befuddles us; in which it is applied imitatively to familiar tasks and ways of doing things. The educational computer is doomed, in other words, to spend some time imitating teachers and books, and, because the copy is never as good as the original, to do their tasks less effectively.

Meanwhile, however, through interaction with the medium, educational users are beginning to comprehend some of its unique capacities. In our book, Computer as Teacher (Siegel and Davis, 1984), we describe this process as the evolution of a concept we call "computer imagination." This concept forms the context for our concluding remarks about SYS IV, of which, we believe, it provides an illustration.

On the one hand, SYS IV is a software technology: it is a powerful, conceptually elegant, flexible, sophisticated management system that does what any management system might do, although we hope it does better some things we have described as being critical to our own view of instruction. On the other hand, however, it is a good idea: it embodies computer imagination, not just in its structure but in its design conception. Its design attempts to reach beyond management as currently understood, exploiting the strengths of the medium to reorient the nature of the instruction it delivers in ways we have mentioned briefly here, and in ways we have not been able to discuss because of limitations of space.

We have seen, for example, how a sophisticated management system can dramatically improve the efficiency of instruction by blending testing and teaching, creating a synergistic effect that increases the efficacy of all components of the instructional system. We have also suggested, admittedly sketchily, the kind of impact a sophisticated management system can have on the formation of instruction. Educators have long desired the capacity to observe instruction closely as it was delivered and to alter it easily, based on such observations and measurements. In print media this goal cannot be realized. Only a really expensive textbook development project can afford the time extensively to field test instruction as the materials are being developed. More often, authors and publishers are forced to rely on informal comments volunteered by teacher-users. And even when the comments prove valuable, implementing them in print is still very costly.

The principal thrust of SYS IV's Implementation Scan Utility, as we have said, is management-oriented. Yet, because the system has been designed with the strengths of the medium in mind, with an eye to the emerging potential of the medium, and not merely to do better a task already done--in short, with computer imagination--its contribution to the realization of the desideratum of fluid, continuous, effective instructional development emerges naturally. The absolutely flexible access to data on student performance made available by the ISU produces a "side effect" of enabling precisely the kind of constant monitoring of student progress that we have described, and it does so without the slightest impairment to the function of SYS IV as a management system. Actually, this potential is no side effect at all; instead, it is the result of CBE designed in a way that permits the medium to change the nature of its own reality. Yet, imagine the eventual impact on instructional design and development when those who develop instruction have routine access to such a process of constant monitoring, and when the instruction, the effectiveness of which they monitor, is not locked in expensive hard copy but is computer-based--soft and instantly accessible!

Important as these "bonuses" are, the general concept they illustrate, and of which SYS IV as a whole is but one example, is much more important in realizing the educational potential of the computer. "More SYS IVs" is only the most limited way in which to describe what we need. "More computer imagination" is the real answer. To achieve that, we need centers that offer, among other things, graduate study in the field of computer-based education, the opportunity to research and to develop educational software, and a forum to discuss developmental and policy issues in CBE.

Computer imagination must emerge on its own, perhaps, but, equipped with the right kinds of tools, facilities, and research atmosphere, CBE professionals can hasten its emergence. As our final thought in a paper delivered to the members of ADCIS, we urge most strongly not merely the development of more "computer imaginative" software but the creation of the conditions under which computer imagination may flourish to actualize the potential we all so deeply desire for computers in education.

References

Siegel, M. A., & Davis, D. M. Computer As Teacher. New York: Random House, 1984.

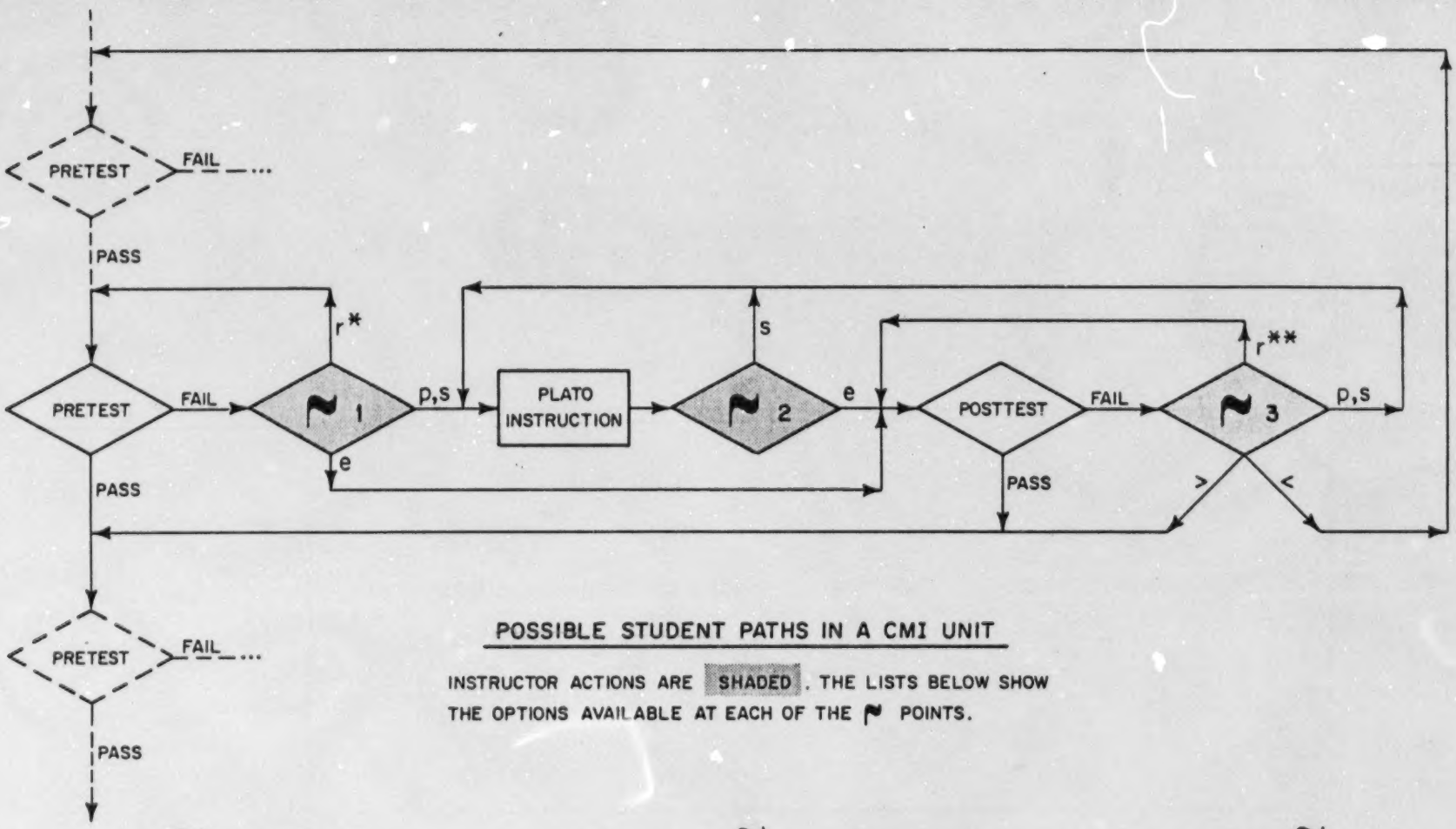
Taylor, R. (Ed.). The Computer in the School: Tutor, Tool, Tutee. New York: Teachers College Press, 1980.

INSTRUCTOR INDEX

Press:	To:
n	Read or write notes
r	Change or inspect records
c	See topic catalog
l	See any PLATO lesson
SHIFT-STOP	Sign off

Your signon last used on 4/05/83 at 10:15 am

Figure 1. The instructor index display.



POSSIBLE STUDENT PATHS IN A CMI UNIT

INSTRUCTOR ACTIONS ARE SHADED. THE LISTS BELOW SHOW THE OPTIONS AVAILABLE AT EACH OF THE ⚑ POINTS.

⚑1

p - PRESCRIPTION
e - ENTER POSTTEST
x - EXTRA OPTIONS:
  s - SPECIAL (ASSIGNMENT)
  r - RETAKE PRETEST *

* ONLY ONE RETAKE OF THE PRETEST IS PERMITTED

⚑2

e - ENTER POSTTEST
x - EXTRA OPTIONS:
  s - SPECIAL (ASSIGNMENT)

⚑3

p - PRESCRIPTION
s - SPECIAL (ASSIGNMENT)
r - RETAKE POSTTEST **
x - EXTRA OPTIONS:
  > - NEXT UNIT
  < - PRIOR UNIT

** RETAKING THE POSTTEST IS PERMITTED ONLY IF STUDENT HAS HAD *PLATO* INSTRUCTION SINCE ENTERING THE CURRENT UNIT

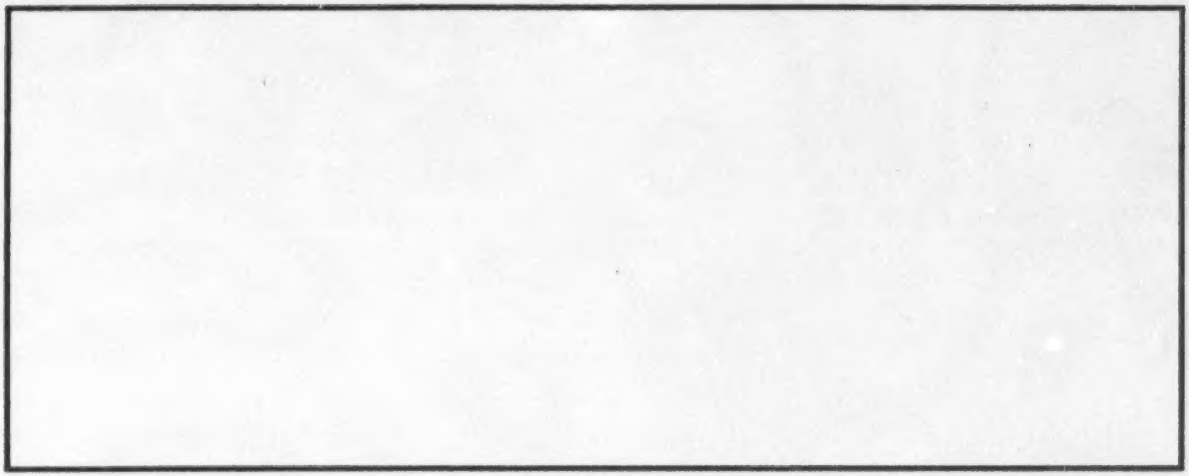
Figure 2. One possible Instructor Logic curriculum structure. This one is used by PCP.

```
Ccc / a:  "a.m. class"                          Remaining Spaces:  9

        Press   a   to add someone to the roster
                d      delete someone from the system
                r      see someone's records
                n      change someone's name
                p      change someone's password

 ------------------------------------------------------------------------

    1  alse<      s (S)        ⚑ 16  leo        5)
    2  andr<      )                17  phil       n (S)
    3  anth<        (S)            18  rayn       nan (S)
    4  berli      n (S)            19  roge       :t (S)
    5  dair<      s (S)            20  stan       :is (S)
    6  derek      5)               21  will       n (S)
    7  elmer        (S)
    8  james        (S)
    9  james        (S)
   10  jerry      n (S)
   11  john
   12  kenne      on (S)
 ⚑ 13  kevin      (S)
   14  larry      on (S)
   15  lawre      : (S)

SHIFT-BACK - Instructor Index
SHIFT-DATA - section options                    LAB - another section
```

Figure 3. The section roster display (last names not shown because these are real students).

Current record for student: bob / ccc / b

1 READING COMPREHENSION
2 MATHEMATICS
3 Synonyms and Antonyms
4 Words Commonly Confused
5 Spelling Rules



g - general info	a - assign	SHIFT-NEXT - next ⚑ record
t - topic summary	d - delete	NEXT - next record
1-5 - single topic	o - order topics	BACK - prior record
c - copy	m - min delay	LAB - roster page
h - history	n - note	SHIFT-HELP - delete record

⚑ indicates student is waiting for an instructor action

Figure 4. The initial student record display (last name not shown). Additional information is displayed in the rectangular box upon instructor request.